Préparation de la séance 2

Objectif: gérer les associations dans l'orm, faire des requêtes sur les associations

Question 1:

```
Categorie(<u>id</u>, nom, descr)
Annonce(<u>id</u>, titre, date, texte)
Photo(<u>id</u>, file, date, taille_octet, #idAnnonce)
CatAnnonce(#idCateg, #idAnnonce)
```

Question 2:

Pour la classe Annonce, il faut utiliser les méthodes hasMany() et belongsToMany() de la manière suivante :

```
public function photos() {
          return $this->hasMany('gamepedia\gp\models\Photo', 'annonce_id');
}
public function categ() {
          return $this->belongsToMany('\gamepedia\gp\models\Categorie',
'categ_annonce', 'annonce_id', 'categ_id'); //Avec : CatAnnonce : le nom de la table qui
fait l'association entre catégorie et annonce, et idAnnonce et idCateg les deux
colonnes de la table
}
```

Pour la classe Photo, il faut utiliser la méthode belongsTo() de la manière suivante : public function annonce() { return \$this->belongsTo('gamepedia\gp\models\Annonce', 'annonce_id'); }

Question 3:

- Requête pour les photos de l'annonce 22 Photo::where('annonce_id', '=', 22)->get();
- Requête pour les photos de l'annonce 22 dont la taille en octets est > 100 000 Photo::where([['annonce_id', '=', 22], ['taille_octet', '>', 100000]])->get();
- 3. Requête pour les annonces possédant plus de 3 photos Annonce::has('photos', '>' 3)->get();
- 4. Requête pour les annonces possédant des photos dont la taille est > 100 000 Annonce::has('photos', function(\$photo) { \$photo->where('taille_octet', '>', 100000); })->get(); });

Question 4:

```
$annonce = Annonce::find(22);
$photo = new Photo();
$photo->id = 365;
$photo->file = 'Corail.jpg';
$photo->date = '2018-03-12';
$photo->taille_octet = 106 496;
$annonce->photos()->save($photo);
Question 5 :
$annonce = Annonce::find(22);
$categ1 = Categorie::find(42);
$categ2 = Categorie::find(73);
$categ1->annonce()->save($annonce);
$categ2->annonce()->save($annonce);
```